Acta Astronautica Vol. 25, No. 3, pp. 119–129, 1991
Printed in Great Britain.

0094-5765/91 \$3.00 + 0.00

AN EXPLICIT CLOSED-LOOP GUIDANCE FOR LAUNCH VEHICLES

R. VIJAYA VITTAL and M. SEETHARAMA BHAT
Department of Aerospace Engineering, Indian Institute of Science, Bangalore 560012, India

(Received 9 May 1990; received for publication 10 December 1990)

Abstract—An explicit, near optimal, closed-loop steering logic common to both, 2- and 3-dimensional launch trajectories is presented. This scheme uses a single steering angle in a canted plane, which can then be resolved as pitch and yaw angles in any inertial frame. New, exact formulations for integrating the steering thrust acceleration are presented along with the existing series approximation. Using Enke's method, two uniform gravity vectors are evaluated to account for the actual effects of inverse square gravitational force field in the propagation of position and velocity vectors, during the future guided trajectory. An iterative differential corrector to solve for the guidance parameters that decide the steering angles and thrust cut-off is discussed. Also, means of overcoming the difficulties faced by the closed-loop guidance logic close to injection is presented. Simulations are conducted for a typical launch mission. The computational load for the onboard computer are estimated in order to explore the feasiblity of realizing this guidance logic. Simulation study of the effects of computational delay is also presented.

1. INTRODUCTION

Satellite launch vehicles are characterized by many uncertainties due to rapid burning of fuel, swift changes in vehicle parameters, high accelerations, discontinuous thrusting of multistage vehicles and changes in the environment. To use such vehicles for accurate orbital injection of the payloads, a highly accurate, optimal, closed-loop guidance (CLG) logic is required. Over the last three decades many guidance schemes have been developed [1–3]. The explicit methods determine in real time the steering commands, based on present and target states. Most of these schemes employ certain optimization criteria for improved performance. Since, real time solutions are essential for explicit guidance schemes, simplifying assumptions are necessary to get analytical solutions for the highly non-linear two point boundary value problem. At each update cycle, the parameters of optimal logic are calculated depending on the current and the target conditions only. The repeated updates of guidance parameters can ensure the attainment of the terminal conditions through redefinition of the optimal trajectory at each cycle. Therefore, the explicit schemes are flexible and capable of overcoming larger uncertainties. They permit redefining of target conditions during the flight itself for the mission salvage. They can also be easily adapted for different missions with the same rocket. Hence, the explicit guidance schemes are preferable for developmental vehicles where the uncertainties are significant.

Developmental efforts in the explicit guidance schemes can be traced back to Lawden's [4] derivation of optimal thrust direction in an analytical form for, uniform gravity, constant thrust and mass flow rate and vacuum conditions of flight. The optimal thrust vector orientation is given by the bilinear tangent law of the form:

$$\tan[\psi(t)] = (A + Bt)/(C + Dt),$$

where $\psi(t)$ is the thrust attitude angle, A, B, C, D are the steering coefficients determined from the boundary conditions and t represents the time variable. When the horizontal range is a free terminal parameter, while terminal velocity and altitude are specified, the coefficient D in the above bilinear tangent law becomes zero, leading to the linear tangent law as:

$$\tan[\psi(t)] = A' + C't.$$

Chandler and Smith [5] extend the linear tangent law to develop a 3-dimensional guidance scheme. The steering logic for the yaw and pitch are given in the decoupled form, under the assumption of uniform gravity. The uniform gravity assumption with the equivalent flat Earth approximation is valid for a short launch trajectory with small magnitudes of vehicle velocity.

Cherry [6] overcomes the limitations due to the uniform gravity field assumption, by judiciously combining the sphericity of earth with the guidance law, as:

$$\sin[\psi_y(t)] = A + Bt - (\mathbf{g} \cdot \mathbf{y})/a_T,$$

$$\sin[\psi_p(t)] = C + Dt - (\mu/R^2 + V_H^2/R)/a_T,$$

where $\psi_y(t)$, $\psi_p(t)$ are the thrust attitude angles in yaw and pitch directions, μ is the gravitational constant, a_T is the thrust acceleration magnitude, V_H is the vehicle velocity along horizontal direction in orbit plane, R is the radial distance of the vehicle projected

on to the orbit plane, **y** is the position vector perpendicular to the orbit plane, **g** is the acceleration due to gravity. Teren [7] uses similar method for removing gravity effects entering into the dynamical equations. It is observed that, as the target is approached, attitude turn rates toward instability due to the numerical errors and requirement of zero injection error to be met in the remaining short time. Teren [7] suggests termination of CLG calculations except for *Tgo* parameter (time to thrust cut off), near the final injection. The thrusting is terminated on attaining the required injection energy level.

Bittner [8] derives a tangent form of steering law by taking into account the coupling between pitch and yaw as shown below:

$$\tan[\psi_y(t)] = A + Bt,$$

$$\tan[\psi_p(t)] = (C + Dt)\cos[\psi_y(t)].$$

He suggests the minimization of the performance index in the form of terminal error and total burn time instead of demanding exact injection as,

$$J = (\mathbf{d} - \mathbf{X}_T)^T[Q](\mathbf{d} - \mathbf{X}_T) + \int_0^{t_f} dt,$$

where **d** is the desired terminal state, $\mathbf{X}_T$ is the estimated injection state and $[Q]$ is a suitable weighing matrix. Thus, the numerical instability close to injection is avoided. The inflight parameter identification is recommended to ease out instability due to the off-nominal performance of launch vehicle. After studying the effects of truncation in on-board computer, Bittner concludes that a 24 bits fixed point arithmetic is necessary.

Lu [9] derives a linear tangent steering law, similar to Marec [10] as:

$$\hat{\lambda}_f(t) = (\hat{\lambda} + \boldsymbol{\lambda}(t - T_\lambda))/[1 + \hat{\lambda}^2(t - T_\lambda)^2]^{1/2},$$

where $\lambda_f(t)$ is the instantaneous unit thrust direction, $\hat{\lambda}$ is the unit reference thrust direction, T_λ is the thrust vector time constant. For 3-dimensional steering also, the vectors $\hat{\lambda}$, $\boldsymbol{\lambda}$ define a canted plane containing the thrust vector. The pitch and yaw angles are obtained from single steering angle within the above canted plane. To get better accuracy, equivalent average gravity vectors $\mathbf{g}_{AVV}, \mathbf{g}_{AVR}$ are separately defined to represent the effects of gravity on estimates of velocity and position vectors respectively.

$$\mathbf{g}_{AVV} = (1/N)\left[\sum_{j=0}^{N} \mathbf{g}_j - (1/2)(\mathbf{g}_0 + \mathbf{g}_N)\right]$$

$$\mathbf{g}_{AVR} = (1/N^2)\left[2\sum_{j=0}^{N} (N - j)\mathbf{g}_j - (N + 1/3)\mathbf{g}_0 + (1/3)\mathbf{g}_N\right]$$

where $\mathbf{g}_j$ is the gravity vector at the jth segment on the future trajectory.

Recently, Sinha *et al.* [11–14] developed a very elegant explicit guidance scheme keeping in view requirements of the Polar Satellite launch vehicle (PSLV). The average gravity vectors are computed using the Enke's method. The guidance scheme has been extended to consider non-uniform mass flow rate and thrusting which are typical of solid propellant engines.

In this paper, further studies on this guidance scheme are attempted. Analytical closed-form evaluation of thrust effects are proposed against approximate evaluations given earlier. Typical on-board computational time requirements are studied.

2. OPTIMAL GUIDANCE LOGIC

The objective of the optimal guidance logic is to determine the thrust attitude angle by a closed-loop action, such that the multistage launch vehicle places the payload/satellite into the desired orbit with minimum thrusting time (time-to-go). The guided trajectory of the vehicle is truly 3-dimensional, since, the plane containing vehicle trajectory at the launch point and that of the final orbit are non-coplanar. The algorithm is tested for the PSLV-class of vehicles.

Launch Point Inertial (LPI) co-ordinates and Orbit Plane Geocentric Inertial (OGI) co-ordinates are used in the analysis. In the LPI co-ordinate system, the axis $\hat{I}_z$ is along the radius joining earth center to the launch point. $\hat{I}_x$ is parallel to launch azimuth and $\hat{I}_y$ completes the right-handed co-ordinate system. The OGI reference frame, shown in Fig. 1, is used to estimate a suitable injection point on the orbit. The five orbital parameters, a, e, i, Ω and θ are specified while leaving argument of perigee (ω) at injection as an unspecified parameter. Then, $\hat{I}_{x0}$ is defined along the projection of the present position vector of the launch vehicle, on the orbit plane and $\hat{I}_{y0}$ is perpendicular to the orbit plane (opposite to the orbital angular momentum vector) $\hat{I}_{z0}$ is the other corresponding co-ordinate of the right-handed system. A range angle θ_T is defined as the angle between $\hat{I}_{x0}$ and the radius vector at the estimated injection point on the orbit.

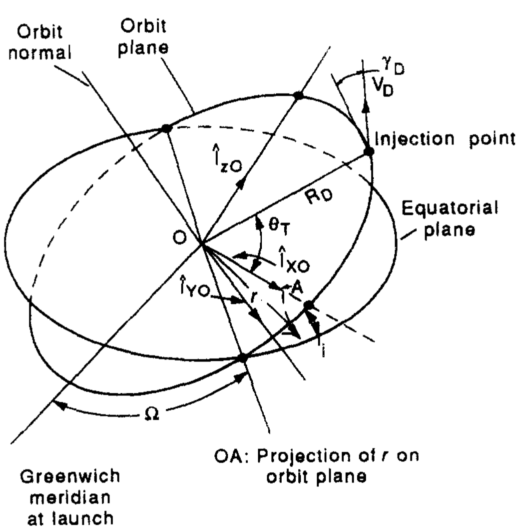

Fig. 1. Orbit plane geocentric inertial co-ordinates.

An estimation of the injection state on the orbit can be obtained using the orbital parameters and the estimate of the range angle θ_T as:

$$\mathbf{V}_T = V_D[\hat{I}_{x0}\sin(\gamma_D - \theta_T) + \hat{I}_{z0}\cos(\gamma_D - \theta_T)],$$
$$\mathbf{R}_T = R_D[\hat{I}_{x0}\cos(\theta_T) + \hat{I}_{z0}\sin(\theta_T)], \quad (1)$$

where V_D,R_D are magnitudes of velocity and radius vectors at specified true anomaly, and γ_D is the flight path angle corresponding to the desired true anomaly at injection.

In order to determine the thrust attitude in real time, the following simplifying assumptions are made:

(1) CLG begins after a short finite period of launch during which, the open loop guidance steers the vehicle beyond land mass constraints and dense atmosphere.

(2) An equivalent constant/uniform gravity vector representation for the vehicle moving around spherical earth is used.

(3) For the remaining part of the flight, nominal parameters of the vehicle defined *a priori*, are used in guidance computations.

The equations of motion for a launch vehicle can be expressed as:

$$\dot{\mathbf{v}} = (F(t)/m(t))\hat{u}(t) + \mathbf{g}, \quad \dot{\mathbf{r}} = \mathbf{v}, \quad (2)$$

where $\mathbf{v}$ and $\mathbf{r}$ are the 3-dimensional velocity and position vectors, $\hat{u}(t)$ is a unit vector along the direction of thrust. $F(t)$, $m(t)$ are the instantaneous thrust magnitude and mass respectively. $\mathbf{g}$ is a constant acceleration due to gravity. Since $\hat{u}(t)$ is a unit vector it satisfies the constraint,

$$\hat{u}(t) \cdot \hat{u}(t) = 1. \quad (3)$$

The vector $\hat{u}(t)$, and the duration of thrust are decided at each instant from the information on current position and velocity vectors from INS (Inertial Navigation System), the predefined target and the fuel optimality. For a rocket with non-throttlable engines, fuel optimality implies minimum time of burn. The associated performance index can be given as,

$$J = \int_0^{t_f} \mathrm{d}t. \quad (4)$$

The necessary conditions include, state and co-state equations and control constraint relations [11, 12] as follows:

$$\dot{\mathbf{P}}_r = 0, \quad \dot{\mathbf{P}}_v = -\mathbf{P}_r \quad (5)$$

where $\mathbf{P}_r$, $\mathbf{P}_v$ are the co-state vectors corresponding to $\mathbf{r}$ and $\mathbf{v}$ respectively. On solving eqn (5) one gets,

$$\mathbf{P}_r = \text{constant}, \quad \mathbf{P}_v = -\mathbf{P}_r t + \mathbf{C}. \quad (6)$$

The transversality conditions involve certain state variables being specified at the terminal point. The remaining adjoint variables are set to zero at the terminal time, since, the performance index does not contain a penalty on the unspecified state variables. The control constraint relation leads to,

$$(F(t)/m(t))\mathbf{P}_v = 2P_u\hat{u}(t), \quad (7)$$

where P_u is the adjoint variable associated with the constraint in eqn (3). Therefore, the unit thrust vector $\hat{u}(t)$ lies in a plane defined by $\mathbf{P}_r$ and $\mathbf{C}$, [eqns (6–7)] even for the 3-dimensional launch trajectory as it is along $\mathbf{P}_v$.

In general, $\mathbf{P}_r$, $\mathbf{P}_v$ can be represented in any inertial frame of reference. Since, the thrust vector lies in a plane throughout its operation, an appropriate guidance frame is defined as shown in Fig. 2. The coordinate $\hat{I}_{yc}$ is defined along $-\mathbf{P}_r$, $\hat{I}_{xc}$ is perpendicular to the $\hat{I}_{yc}$ in the $(\mathbf{P}_r, \mathbf{P}_v)$ plane, and towards $\mathbf{P}_v$. $\mathbf{P}_v$ can now be resolved along $\hat{I}_{xc}$ and $\hat{I}_{yc}$ as:

$$\mathbf{P}_v = C_x[\hat{I}_{xc} + \hat{I}_{yc}P(t - T_c)], \quad (8)$$

where $\mathbf{C} = \hat{I}_{xc}C_x + \hat{I}_{yc}C_y$, $P = P_r/C_x$ and $T_c = C_y/P_r$. Since, $\hat{u}(t)$ and $\mathbf{P}_v$ are collinear, one gets,

$$\hat{u}(t) = [\hat{I}_{xc} + \hat{I}_{yc}P(t - T_c)]/[1 + P^2(t - T_c)^2]^{1/2}. \quad (9)$$

The optimal thrust vector lies in the $\hat{I}_{xc}$, $\hat{I}_{yc}$ plane. Therefore, the changes in position and velocity vectors due to thrusting alone necessarily lie in this plane $(\mathbf{V}_{thR}, \mathbf{R}_{thR})$. Hence, this plane can be called as correction plane. The incremental position vector can be split along $\mathbf{V}_{thR}$ (i.e. $\mathbf{R}_{thm}$) and a direction perpendicular to $\mathbf{V}_{thR}$ (i.e. $\mathbf{R}_{thn}$) in the correction plane (Fig. 2). For launching the payload into the orbit, there is a freedom in terms of injection on the orbit. This enables the $\mathbf{R}_{thm}$ component to be free, since for every choice of $\mathbf{R}_{thm}$, there exists a possibility of achieving the orbit injection requirements. The other component $\mathbf{R}_{thn}$ gets suitably redefined for each choice of $\mathbf{R}_{thm}$. Now, the costate vector $\mathbf{P}_r$ can be split along $\mathbf{R}_{thm}$ and $\mathbf{R}_{thn}$ as:

$$\mathbf{P}_r = \text{unit}(\mathbf{R}_{thm})P_{rm} + \text{unit}(\mathbf{R}_{thn})P_{rn}.$$

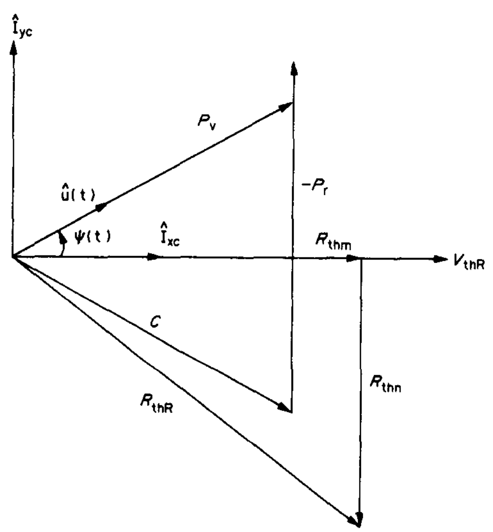

Fig. 2. Correction plane.

Since, $\mathbf{R}_{thm}$ is unspecified, the corresponding transversality condition is,

$$P_{rm}(t_f) = 0.$$

Since, $\mathbf{P}_r$ is a constant vector, its component $\mathbf{P}_{rm}$ is identically zero. Therefore, $-\mathbf{P}_{rn}$ is collinear with $\hat{I}_{yc}$, and hence, $\mathbf{V}_{thR}$ which is along $\mathbf{R}_{thm}$ is collinear with $\hat{I}_{xc}$. The use of a correction plane helps to resolve the contribution due to thrust on position and velocity vectors. From Fig. 2, the optimal steering law is given by the linear tangent law of the form,

$$\tan[\psi(t)] = P(t - T_c). \tag{10}$$

Equation (10) is an important milestone in the development of CLG. The guidance parameters P, T_c, $\hat{I}_{xc}$ and $\hat{I}_{yc}$ in eqn (9) are determined at each instant of guidance computation from the knowledge of the present and target positions and velocities, the contributions due to thrust and gravity. After determining above mentioned parameters, the pitch and yaw commands $\psi_p(t)$, $\psi_y(t)$ and their time rates can be computed with respect to LPI coordinates by projecting $\hat{u}(t)$ on pitch and yaw planes as:

$$\begin{aligned}
\psi_p(t) &= \tan^{-1}[(I_{xcx} + I_{ycx}P(t - T_c))/(I_{xcz} + I_{ycz}P(t - T_c))] \\
\psi_y(t) &= \tan^{-1}[(I_{xcy} + I_{ycy}P(t - T_c))/K], \\
\dot{\psi}_p(t) &= P[(I_{xcz}I_{ycx} - I_{xcx}I_{ycz})/K^2], \\
\dot{\psi}_y(t) &= [(KPI_{ycy} - \dot{K}(I_{xcy} + I_{ycy}P(t - T_c)))/ \\
&\qquad (1 + P^2(t - T_c)^2)],
\end{aligned} \tag{11}$$

where, I_{xcx}, I_{xcy}, I_{xcz}, I_{ycx}, I_{ycz} are x, y, z components of unit vectors $\hat{I}_{xc}$, $\hat{I}_{yc}$ in LPI coordinate frame respectively, and

$$K^2 = [(I_{xcz} + I_{ycz}P(t - T_c))^2 + (I_{xcx} + I_{ycx}P(t - T_c))^2],$$
$$\dot{K} = (P/K)[P(t - T_c)(1 - I_{ycy}^2) - I_{xcy}I_{ycy}].$$

3. IMPLEMENTATION OF THE GUIDANCE LOGIC

To find guidance parameters mentioned above, estimation of velocity and position increments to be gained through thrusting, are required. These requirements can be calculated after estimating the effects of gravity during the future trajectory, and finding a suitable injection point on the orbit. On the other hand, a proper choice of guidance parameters fixes the thrust profile. The corresponding velocity and position increments are then matched with the above requirements.

3.1. *Calculation of thrust effects*

The changes in velocity and position due to available thrust force $F(t)$ acting along optimal $\hat{u}(t)$ direction can be written as:

$$\mathbf{V}_{thA} = \int_0^{Tgo} (F(t)/m(t))\hat{u}(t)\,\mathrm{d}t,$$

$$\mathbf{R}_{thA} = \int_0^{Tgo} \left[\int_0^t (F(s)/m(s))\hat{u}(s)\,\mathrm{d}s\right]\mathrm{d}t, \tag{12}$$

where $\mathbf{V}_{thA}$, $\mathbf{R}_{thA}$ are the velocity and position vector increments though thrusting, and Tgo is the total duration of thrust. These integrals can be evaluated exactly as explained later in Section 3.1.1. One can also use a simpler, but approximate method of evaluation of these integrals using a series expansion for the denominator of $u(t)$ in eqn (9) as:

$$[1 + (P(t - T_c))^2]^{-1/2} = 1 - (1/2)(P(t - T_c))^2 + (3/8)(P(t - T_c))^4 - \ldots . \tag{13}$$

This series converges for $P(t - T_c) < 1$, which implies that eqn (13) is valid for $\psi(t) < 45°$, even though, optimal steering law is valid for $\psi(t) < 90°$, as seen from eqn (10). Using this series as a polynomial in time variable t, $\mathbf{V}_{thA}$, $\mathbf{R}_{thA}$ can be expressed in terms of integrals of the form:

$$Isn_j = \int_0^{Tgo_j} (F_j(t)/m_j(t))t^n\,\mathrm{d}t,$$

$$Idn_j = \int_0^{Tgo_j} \left[\int_0^t (F_j(s)/m_j(s))s^n\,\mathrm{d}s\right]\mathrm{d}t, \tag{14}$$

where, index j corresponds to a particular stage of the multistage vehicle. Tgo_j is the remaining burn time of jth stage. Idn_j can further be expressed in terms of single integrals, on integrating by parts as,

$$\begin{aligned}
Idn_j &= Tgo_j \int_0^{Tgo_j} (F_j(t)/m_j(t))t^n\,\mathrm{d}t \\
&\quad - \int_0^{Tgo_j} (F_j(t)/m_j(t))t^{n+1}\,\mathrm{d}t, \\
&= Tgo_j Isn_j - Is(n+1)_j.
\end{aligned} \tag{15}$$

Equations (14) and (15) are valid for any thrust model. Most of the liquid propellant rocket engines can be modelled as:

$$\begin{aligned}
(F_j(t)/m_j(t)) &= [F_j(t)/\dot{m}_j(t)]/(Tau_j - t)), \\
&= Vex_j/(Tau_j - t).
\end{aligned}$$

where m_{0j} = vehicle mass at present instant t_0, $\dot{m}_j$ = constant mass flow rate of jth stage, $Vex_j = F_j(t)/\dot{m}_j(t)$ is a constant exhaust velocity for jth stage, and $Tau_j = m_{0j}/\dot{m}_j$ is a vehicle time constant updated at each guidance cycle when m_{0j} is changing with t_0. With this thrust model, the integrals in eqn (14) become [7, 11, 12]:

$$Isn_j = \int_0^{Tgo_j} [Vex_j/(Tau_j - t)]t^n\,\mathrm{d}t.$$

On further simplification, the above integrals can be expressed as,

$$Isn_j = -(1/n)Tgo_j^n\,Vex_j + Tau_j Is(n-1)_j, \tag{16}$$

with $Is0_j = Vex_j \ln[Tau_j/(Tau_j - t)]$. Using eqn (16), all thrust integrals can be found sequentially starting with $Is0_j$. These integrals are evaluated at every guidance cycle instant, only for the current stage (j = cs) and the final stage. The integrals for intermediate stages need to be computed only once at the

beginning of the closed-loop guidance, as these do not vary during cycle to cycle computations.

Neglecting third order and higher terms in binomial expansion in eqn (13), relations (12) can be expressed for an N-stage vehicle as,

$$\mathbf{V}_{\text{thAx}} = \hat{I}_{xc}\left[\sum_{j=\text{cs}}^{N}(VAX_{1j} + VAX_{2j})\right],$$

$$\mathbf{0} = \hat{I}_{yc}\left[\sum_{j=\text{cs}}^{N}(VAY_{1j} + VAY_{2j})\right] = \mathbf{V}_{\text{thAy}},$$

$$\mathbf{R}_{\text{thA}} = \hat{I}_{xc}\left[\sum_{j=\text{cs}}^{N}(RAX_{1j} + RAX_{2j})\right] + \hat{I}_{yc}\left[\sum_{j=\text{cs}}^{N}(RAY_{1j} + RAY_{2j})\right],$$

where suffixes x and y indicate $\hat{I}_{xc}, \hat{I}_{yc}$ components, and,

$$VAX_{1j} = Is0_j,$$
$$VAX_{2j} = -(P^2/2)[Is2_j - 2Is1_jT_{cj} + Is0_jT_{cj}^2],$$
$$VAY_{1j} = Is1_j - Is0_jT_{cj},$$
$$VAY_{2j} = -(P^2/2)[Is3_j - 3Is2_jT_{cj} - 2Is1_jT_{cj}^2 + Is0_jT_{cj}^3],$$
$$RAX_{1j} = Id0_j + t_{rj}VAX_{1j},$$
$$RAX_{2j} = -(P^2/2)[Id2_j - 2Id1_jT_{cj} + Id0_jT_{cj}^2] + t_{rj}VAX_{2j},$$
$$RAY_{1j} = Id1_j + t_{rj}VAX_{1j},$$
$$RAY_{2j} = -(P^2/2)[Id3_j - 3Id2_jT_{cj} + 3Id1_jT_{cj}^2 + Id0_jT_{cj}^3] + t_{rj}VAX_{2j}.$$

Here, $T_{cj} = T_c - T_{sj}$, and T_{sj} is the starting time of jth stage with respect to present instant t_0, except for $j = \text{cs}$, for which it is zero. The remaining flight time till the injection, after the completion of jth stage, is represented by t_{rj}.

3.1.1. Exact evaluation of thrust integrals. For evaluating the thrust integrals in eqn (12) exactly, constant thrust, uniform mass flow rate model as explained earlier is used. The effects of thrust can be expressed in terms of the following integrals defined for the jth stage as:

$$I_{xj} = \int_{T_{sj}}^{T_{ej}} Vex_j/[(Tu_j - t)R_t^{1/2}]\,\mathrm{d}t,$$

$$I_{tj} = \int_{T_{sj}}^{T_{ej}} [Vex_j/R_t^{1/2}]\,\mathrm{d}t, \tag{17}$$

where $Tu_j = Tau_j + T_{sj}$, $T_{ej} = T_{sj} + Tgo_j$, $R_t = 1 + P^2(t - T_c)^2$. R_t can be expressed as a polynomial in t as:

$$R_t = c_tt^2 + b_tt + a_t,$$

where $c_t = P^2, b_t = -2P^2T_c, a_t = 1 + T_c^2P^2$. On substituting a new variable x_j defined as $x_j = 1/(Tu_j - t)$, I_{xj} can be expressed in a similar manner to I_{tj} in eqn (17) as:

$$I_{xj} = \int_{x_{j1}}^{x_{j2}} (Vex_j/R_{xj}^{1/2})\,\mathrm{d}x_j.$$

where $R_{xj} = c_{xj}t^2 + b_{xj}t + a_{xj}$, with $c_{xj} = 1 + P^2(Tu_j - T_c)$, $b_{xj} = -2P^2(Tu_j - T_c)$, $a_{xj} = P^2(Tu_j - T_c)$ and $x_{j1} = 1/(Tu_j - T_{sj})$, $x_{j2} = 1/(Tu_j - T_{ej})$. Evidently, c_t and c_{xj} are always positive, they being quadratic functions. Hence, the above standard definite integrals can always be expressed after performing integration, as:

$$I_{xj} = (Vex_j/(c_{xj})^{1/2}\ln\left[2(c_{xj}R_{xj})^{1/2} + 2c_{xj}x_j + b_{xj}\right]\Big|_{x_{j1}}^{x_{j2}}$$
$$I_{tj} = (Vex_j/(c_t)^{1/2})\ln[2(c_tR_t)^{1/2} + 2c_tt + b_t]|_{T_{sj}}^{T_{ej}}. \tag{18}$$

With the above definitions, relations (12) become,

$$\mathbf{V}_{\text{thAx}} = \hat{I}_{xc}\sum_{j=\text{cs}}^{N} I_{xj}$$

$$\mathbf{O} = \hat{I}_{yc}P\sum_{j=\text{cs}}^{N}((Tu_{xj} - T_c)I_{xj} - I_{tj})$$

$$\mathbf{R}_{\text{thAx}} = \hat{I}_{xc}[Tgo\,V_{\text{thAx}} + \sum_{j=\text{cs}}^{N}(I_{tj} - I_{xj}Tu_j)]$$

$$\mathbf{R}_{\text{thAy}} = \hat{I}_{yc}[Tgo\,V_{\text{thAy}} + P\sum_{j=\text{cs}}^{N}[Vex_j((R_t)^{1/2}/c_t)|_{T_{sj}}^{T_{ej}} - (Tu_j - T_c - (b_t/2c_t))I_{tj} - Tu_j(Tu_j - T_c)I_{xj}]]. \tag{19}$$

3.2. Computation of gravity effects

In order to obtain accurate injection, an equivalent uniform gravity field over a flat Earth has been defined [9–12]. Two separate gravity vectors $\mathbf{g}_\text{p}$ and $\mathbf{g}_\text{V}$ are defined such that change in position and velocity along the guided trajectory over a spherical Earth (inverse square field) is approximated to that over the flat Earth as follows,

$$\mathbf{g}_\text{v} = \left[\int_0^{Tgo}(\mu/|r^3(t)|)\mathbf{r}(t)\,\mathrm{d}t\right]\Big/\left[\int_0^{Tgo}\mathrm{d}t\right],$$

$$\mathbf{g}_\text{p} = \left[\int_0^{Tgo}\int_0^{t}(\mu/|r^3(s)|)\mathbf{r}(s)\,\mathrm{d}s\,\mathrm{d}t\right]\Big/\left[\int_0^{Tgo}\int_0^{t}\mathrm{d}s\,\mathrm{d}t\right]. \tag{20}$$

The changes in velocity and position vectors ($\mathbf{V}_g$, $\mathbf{R}_g$) due to gravity are:

$$\mathbf{V}_g = Tgo\,\mathbf{g}_\text{v}, \quad \mathbf{R}_g = (1/2)Tgo^2\,\mathbf{g}_\text{p}. \tag{21}$$

The integral terms on the numerator of eqn (20) can not be evaluated directly as $\mathbf{r}(t)$ depends both on gravity and thrust acting simultaneously. Hence, $\mathbf{V}_g$, $\mathbf{R}_g$ are found using Enke's method [14]. In this method an osculating reference trajectory is established, using only central force motion. The increments in the state variables (position and velocity)

due to thrust are added to the propagated state on the osculating trajectory at the rectification points. Since, thrust integrals are known for individual stages, the burnout instants of each stage can be the natural rectification points. On propagating state from the time t_0 for a time step of $\Delta t = t_1 - t_0$, using an osculating trajectory one gets $\mathbf{r}_1$, at time t_1 (Fig. 5). The correction $\delta\mathbf{r}$ is defined as the difference between actual position vector $\mathbf{r}$ at t_1 and propagated position vector $\mathbf{r}_1$, i.e.

$$\delta\mathbf{r}(t_1) = \mathbf{r}(t_1) - \mathbf{r}_1.$$

The $\delta\mathbf{r}$ can be thought of as resulting from accelerations due to perturbing forces along the osculating trajectory. Such an acceleration can be given as,

$$\delta\ddot{\mathbf{r}}(t_1) = \ddot{\mathbf{r}}(t_1) - \ddot{\mathbf{r}}_1.$$

At all rectification points, the initial states of the next osculating trajectory are corrected to the true state of the vehicle. Hence, $\delta\mathbf{r}$ is zero at these instants. It increases gradually as one moves along the osculating trajectory. At each rectification point, the correction $\delta\mathbf{r}$ is computed in stages, i.e. $\delta\mathbf{r} = \delta\mathbf{r}_1 + \delta\mathbf{r}_2 \cdots$. The first correction $\delta\mathbf{r}_1$, which is dominant, is due to the application of the thrust vector for the duration Δt. In reality the gravitational and thrust forces act simultaneously. The perturbations $\delta\mathbf{r}_2, \delta\mathbf{r}_3, \cdots$ due to coupling between the above two forces are computed sequentially [14] until the subsequent corrections, $\delta\mathbf{r}_i$, are small. The oblateness effects can also be considered, if very high accuracy is desirable. The state variables are similarly obtained at all rectification points until the thrust cutoff where the state estimates (i.e. $\mathbf{V}_{Te}, \mathbf{R}_{Te}$) thus obtained are generally quite close to the desired target conditions. Then, $\mathbf{g}_v$, $\mathbf{g}_p$, are computed as,

$$\mathbf{g}_v = (\mathbf{V}_{Te} - \mathbf{V}_0 - \mathbf{V}_{thA})/Tgo.$$

$$\mathbf{g}_p = 2(\mathbf{R}_{Te} - \mathbf{r}_0 - \mathbf{V}_0\,Tgo - \mathbf{R}_{thA})/Tgo^2. \tag{22}$$

3.3. Solution to guidance parameters

The solution to the guidance problem consists of finding the parameters that decide the thrust steering angles and the time of thrust cutoff such that following constraints are satisfied:

$$\begin{aligned} \mathbf{V}_{thR} &= \mathbf{V}_{thAx}, \\ \mathbf{0} &= \mathbf{V}_{thAy}, \\ \mathbf{R}_{thR} &= \hat{I}_{xc} R_{thAx} + \hat{I}_{yc} R_{thAy}, \end{aligned} \tag{23}$$

where the required velocity and position $\mathbf{V}_{thR}, \mathbf{R}_{thR}$ are defined as,

$$\begin{aligned} \mathbf{V}_{thR} &= \mathbf{V}_T - \mathbf{V}_0 - \mathbf{g}_v\,Tgo, \\ \mathbf{R}_{thR} &= \mathbf{R}_T - \mathbf{R}_0 - (1/2)\mathbf{g}_p\,Tgo^2 - Tgo\;\mathbf{V}_0, \end{aligned} \tag{24}$$

The right-hand side of eqn (24) depends on values of P, T_c, Tgo, $\hat{I}_{xc}$ and $\hat{I}_{yc}$. The gravity vectors $\mathbf{g}_v$, $\mathbf{g}_p$, in eqn (24) also depend on all these parameters. $\mathbf{V}_T$, $\mathbf{R}_T$ depend on Tgo and θ_T. Solving eqns (23) and (24) for the guidance parameters is very cumbersome, as these equations are highly nonlinear. These parameters do not vary by a large quantity from one guidance cycle to the next. Therefore, it is possible to estimate gravity contributions fairly accurately from an earlier knowledge of the parameters, since the sensitivity of such an estimate is not large. The guidance parameters are refined using differential corrections while holding $\mathbf{g}_v, \mathbf{g}_p$ constant. However, at the initiation of the closed-loop guidance, one may not have reasonable estimate of parameters to compute $\mathbf{g}_v, \mathbf{g}_p$ to the requisite accuracy. Therefore, the above two-step procedure is repeated.

3.3.1. Partial differential corrector. The vectorial constraints (25) are reformulated as scalar relations for the sake of simplicity as,

$$F_1 = \mathbf{V}_{thR} \cdot \mathbf{V}_{thR} - \mathbf{V}_{thAx} \cdot \mathbf{V}_{thAx} \tag{25a}$$

$$F_2 = \mathbf{R}_{thR} \cdot \mathbf{R}_{thR} - \mathbf{R}_{thA} \cdot \mathbf{R}_{thA} \tag{25b}$$

$$F_3 = \mathbf{V}_{thR} \cdot \mathbf{R}_{thR} - \mathbf{V}_{thA} \cdot \mathbf{R}_{thA} \tag{25c}$$

$$F_4 = \mathbf{V}_{thAy} \cdot \mathbf{V}_{thAy}. \tag{25d}$$

By the proper choice of Tgo, P, and T_c, F_1 to F_4 can be driven to zero. T_c is initially solved using the expression for F_4. Then, Tgo, P, θ_T are solved simultaneously by a differential corrector method using eqns (25a–c). With these, $\mathbf{V}_{thR}$ and $\mathbf{R}_{thR}$ are updated, and $\hat{I}_{xc}, \hat{I}_{yc}$, are computed using,

$$\begin{aligned} \hat{I}_{xc} &= \text{Unit vector } (\mathbf{V}_{thR}), \\ \hat{I}_{yc} &= \text{Unit vector } (\mathbf{R}_{thR} \times \hat{I}_{xc}) \times \hat{I}_{xc}. \end{aligned} \tag{26}$$

Finally, the corrections to T_c as a result of changes in Tgo, and P, is provided by the relation,

$$\mathrm{d}T_c = (\partial T_c/\partial Tgo)\,\mathrm{d}Tgo + (\partial T_c/\partial P)\,\mathrm{d}P. \tag{27}$$

Here, partial derivatives of T_c are obtained from relation in eqn (25d). The required partial derivatives are evaluated only once in a guidance cycle except during the first cycle on ignition of each stage. This reduces guidance computations substantially without sacrificing accuracy or optimality. This is so because, subsequent corrections are very small to cause appreciable changes in partial derivatives.

3.3.2. Starting estimates of guidance parameters. Partial differential corrector methods described above need good starting estimates of the guidance parameters for fast convergence. Only the Tgo estimate based on ground simulations needs to be supplied externally. All other parameters can be estimated onboard at the starting of the CLG. However, for each subsequent guidance cycles, simple starting update formulae can be derived using the parameters from the previous cycle.

Initial estimates at the start of CLG: The first approximation to range angle θ_T is obtained by averaging the angular velocity of the vehicle at the

guidance initiation time T_0 and the orbital angular velocity at injection, using *Tgo* estimate, as,

$$\theta_T = (1/2)[(\mathbf{V}_0 \cdot \hat{I}_{z0})/(\mathbf{R}_0 \cdot \hat{I}_{x0}) + (V_D \cos \gamma_D)/(R_D)]Tgo. \quad (28)$$

Here, V_D, γ_D, R_D can be computed from the orbital parameters. $\hat{I}_{x0}, \hat{I}_{z0}$ can then be calculated with the additional knowledge of position and velocity vectors at T_0. Now $\mathbf{V}_T, \mathbf{R}_T$ can be found using relations in eqn (1). Using the present position and estimated injection position, the gravity related vectors are computed from a two-point approximation as,

$$\mathbf{g}_v = (1/2)(\mathbf{g}(\mathbf{R}_0) + \mathbf{g}(\mathbf{R}_T)),$$
$$\mathbf{g}_p = (1/3)(2\mathbf{g}(\mathbf{R}_0) + \mathbf{g}(\mathbf{R}_T)). \quad (29)$$

From the above approximations, $\mathbf{V}_{thR}$, $\mathbf{R}_{thR}$ are calculated [eqn (24)] and in turn $\hat{I}_{xc}, \hat{I}_{yc}$ are found as explained earlier [eqn (26)]. Then considering only the first term in the binomial expansion (eqn. (13)), estimates of T_c and P are obtained as,

$$T_c = \left(\sum_{j=cs}^{N} (Is1_j + T_{sj} Is0_j) \right) \Bigg/ \left(\sum_{j=cs}^{N} Is0_j \right)$$

$$P = -(\mathbf{R}_{thR} \cdot \hat{I}_{yc}) / \sum_{j=cs}^{N} [(Id1_j - Id0_j(T_{sj} + Is1_j/Is0_j)) + t_{rj}(Is1_j + Is0_j(T_{sj} + Is1_j/Is0_j))].$$

Starting approximations at guidance cycles:

At subsequent guidance cycles, the first estimates of guidance parameters are made from their values in previous cycles. *Tgo* and θ_T at the present instant are given by,

$$Tgo = Tgo_{(old)} - \Delta t, \quad \theta_T = \theta_{T(old)} - \Delta\theta_T,$$

where Δt is the guidance cycle interval, and $\Delta\theta_T$ is the angle between projections of present position and previous position on the orbit plane. The value of P is retained as in the previous cycle. Gravity related vectors are also obtained similarly, by,

$$\mathbf{g}_v = [\mathbf{g}_{v(old)} Tgo_{(old)} - \Delta \mathbf{V}_g]/Tgo.$$
$$\mathbf{g}_p = 2[(\mathbf{g}_{p(old)} Tgo^2_{(old)})/2 - \Delta \mathbf{V}_g Tgo - \Delta \mathbf{R}_g]/Tgo^2,$$

where

$$\Delta \mathbf{V}_g = (1/2)[\mathbf{g}(R_{0(old)}) + \mathbf{g}(R_0)](\Delta t)$$
$$\Delta \mathbf{R}_g = (1/6)[2\mathbf{g}(R_{0(old)}) + \mathbf{g}(R_0)](\Delta t^2).$$

The guidance scheme is very sensitive to gravity estimates. Depending on injection accuracy required $\Delta \mathbf{V}_g$, $\Delta \mathbf{R}_g$ can be found by orbital propagation backward in time for the guidance cycle interval. But this leads to increased computational load.

3.3.3. Terminal computations. As the launch vehicle approaches injection point, very large turn rates are demanded if all the CLG computations are carried out until the end. This tendency towards instability results from the vehicle modelling errors, and the demand of the zero injection error to be met in a very short time to go (*Tgo*). In order to overcome such undesirable effects, guidance parameter updates are stopped well before thrust cut off and previously computed parameters are used till the end. Only the velocity magnitude constraint equation [eqn (25a)) is solved to find the *Tgo* accurately. The faster updates of *Tgo* using simpler computations are necessary as the injection accuracy in terms of eccentricity depends greatly on accurate thrust cutoff.

4. SIMULATION STUDY

The simulation study of the CLG logic is made, using typical launch vehicle data (Table 1). The launch vehicle consists of four stages with a relatively long coast phase before the ignition of the final stage. The position and velocity vectors at the start of CLG are also given in Table 1. The parameters of the desired 900 km sun-synchronous orbit are given in Table 2.

The average constant thrust and constant mass flow rate model of each stage in the vehicle are used in CLG computations. Simulation of the dynamical system is made using Adam's Bashforth method, with a step interval of 0.1 s. The guidance scheme consists of major and minor cycles. The duration of each major cycle is 4 s and it consists of four minor cycles of 1 s each. At the beginning of each major cycle all guidance parameters are recomputed. The pitch and yaw angles and their rates [eqns (11)] at each minor cycle are computed using these parameters. In the terminal phase the guidance computations are simplified with the updating of only *Tgo* at a major cycle of 0.5 s. The terminal phase starts when $Tgo \leqslant 5$ s. The thrust integrals are computed using the series

Table 1. Vehicle characteristics and its state at CLG initiation

CLG starting time (with respect to launch instant)	= 204.4 s		
Tgo estimate at CLG start-up	= 795 s		
Vehicle state at (LPI) CLG start-up	Position (km)		
	$X = 285.96$	$Y = 12.96$	$Z = 6550.4$
	Velocity (km/s)		
	$X = 2.51$	$Y = -0.89$	$Z = 1.10$
Launcy vehicle data	2nd stage (liquid)	3rd stage (solid)	4th stage (liquid)
(a) Ignition time (s) (with respect to CLG initiation)	−104.4	50.1	481.9
(b) Total burn time (s)	153.5	91.75	400.0
(c) Exhaust vel (m/s^2)	2860.3	2857.4	3017.7
(d) Initial mass (kg)	54589.6	11908	3696
(e) Mass-flow rate (kg/s)	244.3	79.5	4.6
	coast 1		coast 2
(a) Initiation time (s) (with respect to CLG initiation)	49.1		141.9
(b) Duration (s)	1		340

Table 2. Desired and achieved target conditions

Target conditions	a (km)	e ($\times 10^{-7}$)	i (deg)	Ω (deg)	Altitude (km)	Velocity (km/s)	Last stage burn time (s)
Specified orbit	7278.137	0	98.99	263.76	900	7.400	—
Case (1)	7278.140	3.9312	98.99	263.76	900.001	7.40046	383.78
Case (2)							
Thrust Hi	7278.173	612.23	98.99	263.76	900.000	7.40048	374.28
Lo.	7278.127	574.74	98.99	263.76	900.001	7.40045	393.59
Case (3)	7278.140	2.5266	98.99	263.76	900.001	7.40046	383.32
Case (4)	7278.140	2.2451	98.99	263.76	900.002	7.40046	383.67
Case (5)							
(1) Series integrals	7278.140	2.1465	98.99	263.76	900.002	7.40046	383.94
(2) Exact integrals	7278.140	2.1465	98.99	263.76	900.002	7.40046	383.94

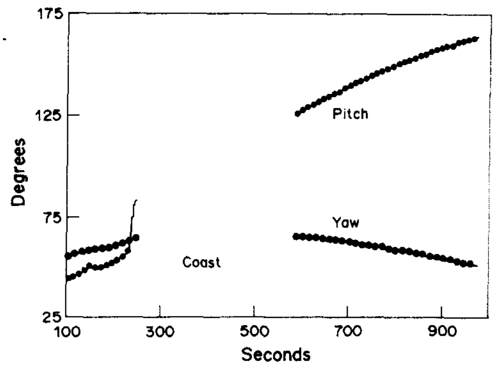

Fig. 3. Pitch and yaw vs time (nominal thrust).

method in cases 1–3, and analytical formulations in case 4 of the simulation studies listed below:

(1) Simulation model of the vehicle is identical to the realistic data.

(2) Off-nominal simulations use uniformly 1% higher or lower thrust value in simulation model alone.

(3) Computational decay effects are simulated. The delay is assumed to be one second uniformly till the terminal phase. At the terminal phase delay is assumed to be negligible.

(4) Simulation data are similar to case (1).

Simulation results indicate, that very high accuracy of injection is possible. The results in the form of demanded pitch and yaw angles, the injection accuracy, are given in Figs 3–7, and Table 2, for all the

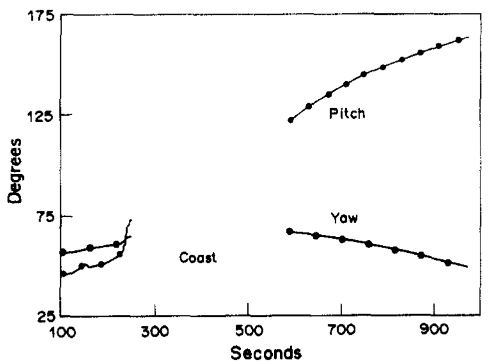

Fig. 4. Pitch and yaw vs time (exact integrals).

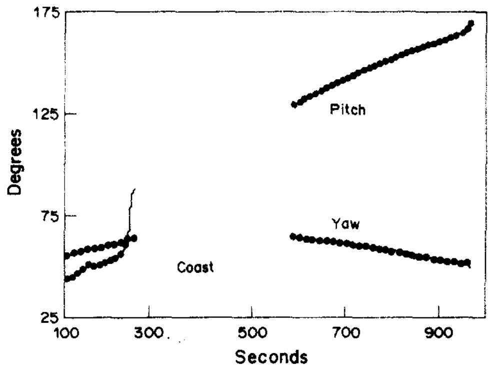

Fig. 5. Pitch and yaw vs time (1% high thrust).

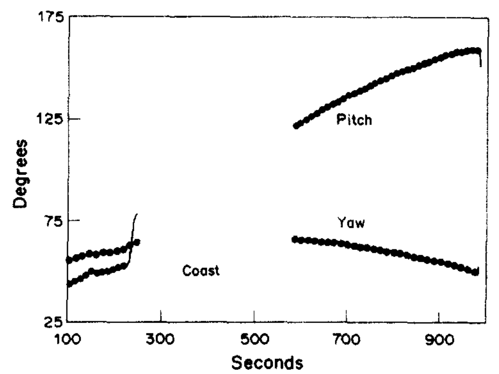

Fig. 6. Pitch and yaw vs time (1% low thrust).

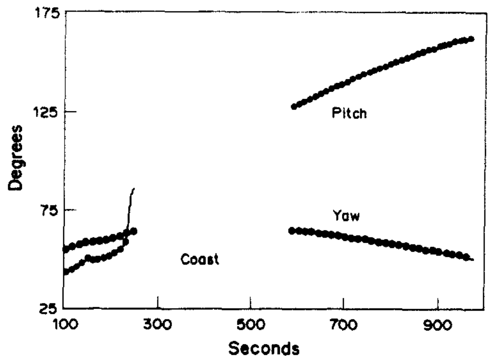

Fig. 7. Pitch and yaw vs time (1 s delay).

cases of simulations done above. The altitude and velocity profiles and all the guidance parameters are also plotted (Figs 10–13). The thrust steering angles in pitch and yaw vary smoothly over the entire trajectory with the exception of a short period at the end of penultimate solid propellant stage. The substantial difference between the average thrust model used in CLG computations and the actual solid propellant model used in simulation is responsible for such large variations in pitch and yaw. It can be seen from Figs 3 and 4, that such variations are small if analytical formulation of thrust integrals are used instead of series approximation method. This is so because, the errors in the series approximation is very high at these instants as $\psi(t)$ will be close to 45° [eqn (10)] [when $\psi(t)$ is 45° the series expansion in eqn (13) diverges]. This difficulty can be overcome by using the guidance parameters from the previous cycles, when the demanded steering angles increase over the previous commands, beyond a reasonable range. The simulation results using such a strategy are shown in Figs 8 and 9, and Table 2, as case (5). This consequently needs a small extra burn time. The better approach, however, is to use the exact thrust model in the CLG computations, at the cost of increased computational time.

On-board computational time requirements are estimated by measuring the time required for the same CLG computations to run repeatedly a large number of times. The average computation time can be found for the guidance cycle computations at

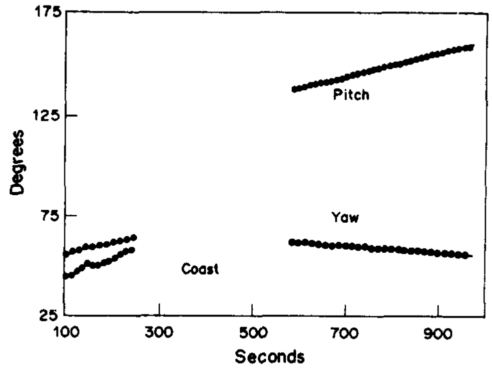

Fig. 8. Pitch and yaw vs time (frozen prmts, series integrals).

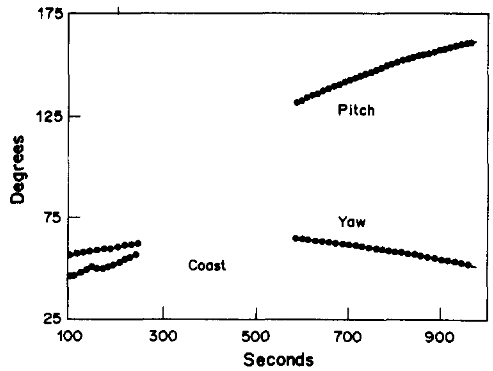

Fig. 9. Pitch and yaw vs time (frozen prmts, exact integrals).

Table 3. Computational requirements

CLG type		Series integration			Analytical integration		
(1) Code memory (kbytes)		31			32		
(2) Data memory (kbytes)		3.5			4		
(3) Execution time							
		Execution time (ms)		No. of diff. corrector	Execution time (ms)		No. of diff. corrector
Stage	CLG cycle time (s)	AT-286	AT-386	iterations	AT-286	AT-386	iterations
2nd	0.0 (CLG start cycle—all computations done twice)	996	165.1	5+4	1682	230	6+4
	8	387	64	3	518	88	4
	16	361	60	3	454	77	3
	20	361	60	2	389	66	2
3rd	52 (stage start cycle)	361	60	3	340	58	2
	96	309	51	2	270	45	1
4th	482.82 (stage start cycle)	253	42	3	302	52	3
	486.82	208	35	2	223	38	2
	494.82	167	28	1	189	32	1
	863.32 (terminal phase cycle)	18.3	3.3	1	18.9	3.3	1

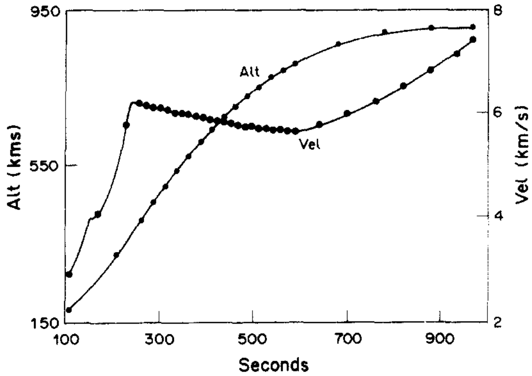

Fig. 10. Altitude and velocity vs time (series integrals).

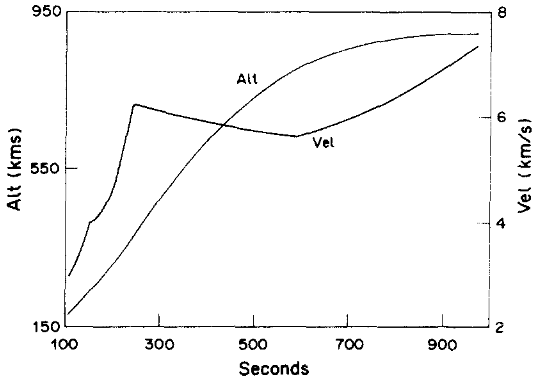

Fig. 11. Altitude and velocity vs time (exact integrals).

various representative instants along the powered trajectory. The following machines are used to run the CLG computations.

(1) IBM compatible (Prakruthi AT/386) PC/AT-386 with 80386 Microprocessor running at 33 MHz with a coprocessor 80387 running at 25 MHz.

(2) IBM compatible (Siva AT) PC/AT-286 with 80286 Microprocessor running at 12 MHz with 80287 coprocessor running at 10 MHz.

The guidance calculations are compiled in Turbo Pascal3 compiler. The typical compiled code and memory requirements are given in Table 3. Since the final implementation can use dedicated hardware and software, the computational requirements indicated here are very conservative. The results of these measurements are given in Table 3.

5. CONCLUSIONS

A closed-loop, explicit, highly accurate and near optimal guidance scheme is given. This scheme is capable of steering the launch vehicles for sophisticated missions, which require large pitch and yaw manoeuvres and long range of trajectories. The explicit scheme requires estimation of effects due to gravity and thrust from the present time till injection along the guided trajectory for determining the guidance parameters and subsequently the steering angles. The gravity effects are computed using Enke's method. The thrust integrals can be evaluated using the series approximation as well as the analytical formulation. Since, the effects due to gravity and thrust are related to the guidance parameters, a sequential algorithm is developed to determine the guidance parameters. Here, the computational time required with the exact analytical formulation is larger by approx. 10–35% compared to that with the series approximation method that uses only the first two terms in the expansion. The accuracy of target injection on the desired orbit is high. The injection errors in the semimajor axis lies in the range of few meters and in eccentricity of the order of 10^{-7}–10^{-5}. The algorithm is computationally feasible for real time implementation. The computational delay does not affect the performance much. The CLG computations can be performed in reasonable time duration (Table 3).

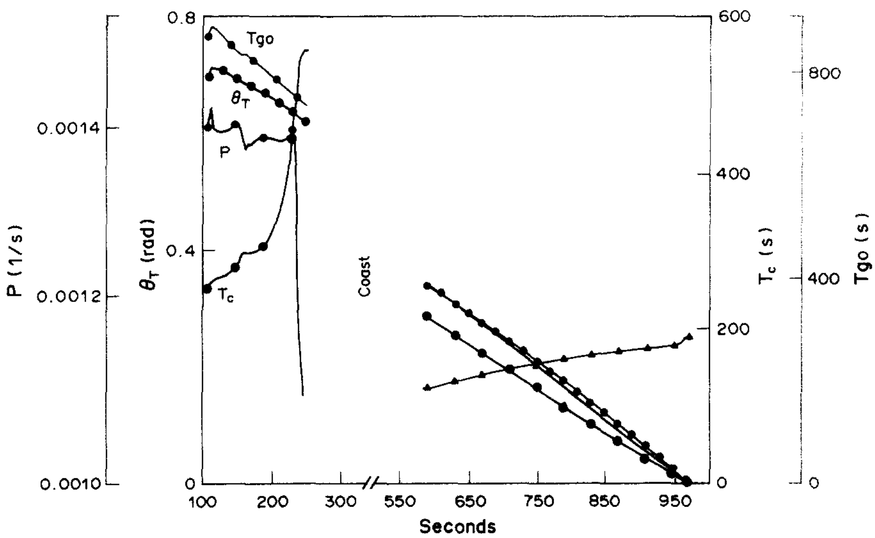

Fig. 12. Guidance parameters vs time (series integrals).

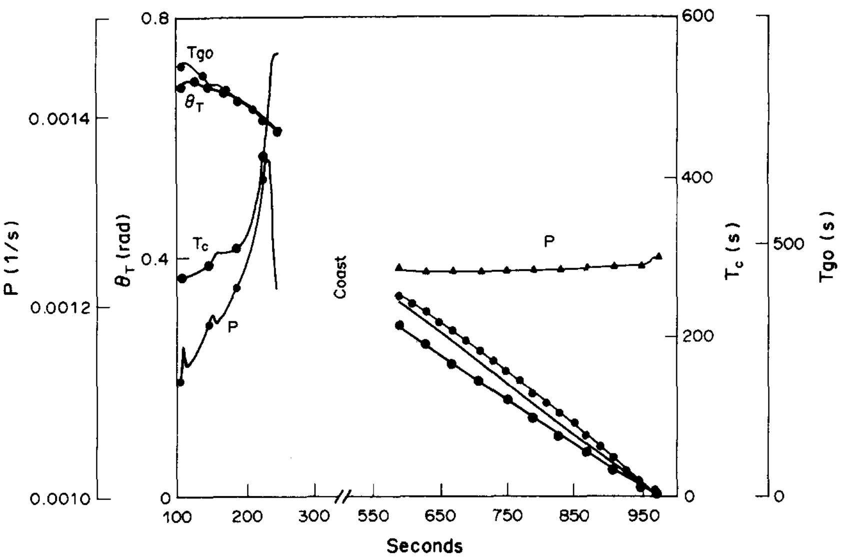

Fig. 13. Guidance parameters vs time (exact integrals).

Acknowledgements—The authors thank Dr S. K. Sinha, ISRO Headquarters, Bangalore for many helpful discussions and valuable suggestions for developing an efficient algorithm.

REFERENCES

1. G. E. Townsend, A. S. Abbot and R. R. Palmer, Guidance flight mechanics and trajectory optimization, Vol. 8: boost guidance equations. NASA CR-1007 (1968).
2. S. K. Shrivastava, M. S. Bhat and S. K. Sinha, Closed loop guidance for satellite launch vehicles, a literature survey'. Report No. IISc 84-FDC 01-15, Indian Institute of Science, Bangalore (1984).
3. S. K. Shrivastava, M. S. Bhat and S. K. Sinha, Closed loop guidance of satellite launch vehicle—an overview. *J. Inst. Engrs India* **66**, 62–76 (1986).
4. D. F. Lawden, *Optimal Trajectories for Space Navigation*. Butterworth, London (1963).
5. D. S. Chandler and I. E. Smith, Development of the iterative guidance mode with its application to various vehicles and missions. *J. Spacecraft Rockets* **4**, 898–903 (1967).
6. G. W. Cherry, A class of unified explicit methods for steering throttlable fixed thrust rockets. *Guidance and Control II*, *Prog. Astronaut. Aeronaut.* **13**, 689–725 (1964).
7. F. Teren, Explicit guidance equations for multistage boost trajectories. NASA-TND-3189 (1966).
8. H. Bittner, Flat-earth guidance law using inflight vehicle parameter identification. *Automatica* **12**, 427–443 (1976).
9. T. C. Lu, An optimal closed-loop, multistage ascent guidance. Rockwell International (1977).
10. J.-P. Marec, *Optimal Space Trajectories*, Chap. 5. Elsevier, New York (1979).
11. S. K. Sinha, Optimal explicit guidance for 3-dimensional trajectory of launch vehicles. Ph.D. thesis, Department of AE, Indian Institute of Science, Bangalore (1987).
12. S. K. Sinha, S. K. Shrivastava, M. S. Bhat, and K. S. Prabhu, Optimal explicit guidance for three-dimensional launch trajectory. *Acta Astronautica* **19**, 115–123 (1989).
13. S. K. Sinha and S. K. Shrivastava, Explicit guidance of a rocket with non-uniform thrust. *Acta Astronautica* **19**, 555–560 (1989).
14. R. R. Bate, D. D. Muller and J. E. White, *Fundamentals of Astrodynamics*, pp. 390–396. Dover, New York (1971).